A SEMI-AUTOMATIC METHOD FOR NEURON CENTERLINE EXTRACTION IN CONFOCAL MICROSCOPIC IMAGE STACK

Ping-Chang Lee[1], Yu-Tai Ching[1], H. M. Chang[2], Ann-Shyn Chiang[2]

[1]Department of Computer Science, National Chiao Tung University, Hsin-Chu, Taiwan
[2]Brain Research Center, National Tsing Hua University, Hsin-Chu, Taiwan

ABSTRACT

An algorithm for extracting the centerlines of neurons from 3-D image stack collected from a laser scanning confocal microscope is presented. Recovery of neuronal structure from image stack is critical for quantitative analysis of neuron-morphology. Many methods have been proposed to extract the centerline from the tubular structure in medical images, such as vessels. But the same methods do not work well in processing of neurons. One of the reasons is that the physical limitations of the spatial resolution of the image stack collect by confocal microscopy in the *z*-direction is much worse than the resolution in *x* and *y* directions. In our studied cases the mean voxel size of the image stack is $0.17 \times 0.17 \times 1.0 \mu m^3$ i.e., the resolution in *z*-direction cannot reflect the fact that the neuron has a tubular structure. In this paper, we propose an almost automatic neuron-tracing algorithm for a set of confocal microscopic images of neuron. The method is designed based on finding a minimal path in the volume to extract the centerlines of a neuron.

Index Terms— confocal microscopy, projection neuron, neuronal structure, centerline extraction, minimum path finding

1. INTRODUCTION

Current progress in imaging technologies have made possible to acquire a 3D image for the distribution of neural fibers in the brain (e.g. *Drosophila*) at high resolution [1][2]. To have a clear anatomical structure of neural networks is critical for the understanding of their functions such as the formation of memory. However, there are more than 100 thousand neurons in the *Drosophila* brain. The task of tracing the circuit for each neuron manually is utterly impractical and eventually impossible. Although there has been some progress in tracing branches of blood vessels with computer algorithms, there has been no similar algorithm for tracing neural processes due to the highly variable morphologies in neurites. Bulges in the middle or at neural terminals are features fundamentally different from blood vessels. New computational algorithms are in need to make an automated tracing for neural fine structures possible. This is especially worthwhile when considering most functions are related to the fine structural modifications at neural terminals.

The methods for tracking line structure for vasculature or anatomical structure can be categorized into four different approaches. The first is based on skeletonization and branch analysis [3]. The second is based on enhancing line or edge properties and then chains up those most likely pixels [4]. The third is referred to vectorial tracking [5]. The fourth is based on minimal cost path finding [6]-[8]. Many of these methods consider the cases that the image resolution in every direction is almost the same or assume that the intensity varies small in the region of interest (at least in a small section of a "tube"). However, these assumptions do not all hold for the *Drosophila's* neuron image derived by confocal microscopy.

In this paper, we propose an almost automatic centerline extracting method for a stack of confocal microscopic images. The proposed method is designed based on the minimal path finding and it is easy to implement. We present the method in next section. The results are shown in Section 3 and we have a discussion in Section 4.

2. METHODS

2.1 Background – The global minimal paths

The minimal path technique proposed in [9] captures the global minimum curve of a contour depending on energy between two given points. The well-known snake model [10] simultaneously considered the smoothness of the curve and the potential term, that was determined by the image features, in the energy functional.

$$E(C) = \int_{\Omega} \{\alpha \|C'(s)\|^2 + \beta \|C''(s)\|^2 + \lambda P(C(s))\} ds \qquad (1)$$

In equation (1), α, β, and λ are real positive constants, $C(s) \in \Re^n$ represents a curve drawn in the image stack, $\Omega = [0, L]$ is its domain of definition where L is the length of the curve, $C'(s)$ and $C''(s)$ are first and second derivative with respect to s and $P(C(s))$ is the potential which is used to capture the desirable image feature. If s

represents the arc-length parameter then we can have a simplified energy form [9]

$$E(C) = \int_{\Omega} \{w + \lambda P(C(s))\} ds = \int_{\Omega} \tilde{P}(C) ds \qquad (2)$$

In this model w is a real positive constant which controls the smoothness of the contour [9] and $\tilde{P} = w + \lambda P$.

Given a potential $P > 0$ that is defined to be small when the contour is close to the desired boundary. The objective of minimal path technique is to look for a path connecting a given pair of points such that the integral of $\tilde{P} = w + \lambda P$ is minimal. A minimal energy action map $U_{p_0}(p)$ is defined as the minimal energy integrated along a path between a point p_0 and another point p in the images

$$U_{p_0}(p) = \inf_{A p_0 p} E(C) = \inf_{A p_0 p} \{ \int_{\Omega} \tilde{P}(C(s)) ds \} \qquad (3)$$

where $A_{p_0 p}$ is the set of all paths between p_0 and p. Once the minimal energy action map is built, the minimal path between p_0 and p can be deduced easily. The minimal path approach has several advantages such as finding the global minimizer and ease of implementation. In the following, we present the proposed method, a 3-D approximated minimal path finding method, based on the global minimal path finding.

2.2 Approximate minimal path method

2.2.1. Pre-processing

Since the spatial resolution of confocal microscopic images in the *z*-direction is much worse than the resolution in *x* and *y* directions, we pre-process the images slice-by-slice first. We then compute the minimal energy map from the processed image stack.

For every slice we choose a threshold based on its intensity histogram and then binarized the image slice. A refined binary image stack, V_b is then obtained. A 3-D 26-neighbor connected component analysis is applied. In most of the cases, the largest connected component is the desired neuron. Let V be the volume containing the binarized neuron. The Euclidean distance transform is then applied to each image slice in V and construct the skeletons, S_k of every object in the foreground of slice k. For each slice k, we compute a set of candidate 3-D end points by examining 9 digital planes in the 26-neighborhood of each end point of S_k. The details of the algorithm are referred to [11]. The set of skeleton points in each slice plays an important role in designing the potential function. The set of candidate end points is denoted E^0.

2.2.2. Awarding function and minimal path deduction

In order to make the path lies in the center of the desirable structure, we define the potential as an *awarding function f* as follows. V can be considered as a grid graph that the vertices are voxels and the edges are defined by the 26-neighborhood in V. $\forall p \in V$ and its neighbor q, there is an edge defined by the pair of connected vertices (p,q), $f(p,q)$ satisfies the conditions:

1) $f(p,q) < 0$ if $q \in S_k$, for some k, otherwise, it equals 0

2) Let $Dis_{Euclidean}(p,q)$ be the Euclidean distance between p and q. $\lambda |f(p,q)| < Dis_{Euclidean}(p,q)$, $\forall q \in S_k$, for some k.

Under the second restriction, we can guarantee that there are no negative edges in the weighted grid graph of V.

By applying the awarding function to deduce the minimal path from a given source point, s is as follows. From the given source point, s, we apply the well-known Dijkstra's algorithm to calculate the single source shortest paths to all the other end points, $t_j \in E^0$ .We iteratively perform the following steps. After each iteration, we remove some candidate end points from E^i to form the E^{i+1}, $i \geq 0$.

1. We pick the longest path among all of the minimal paths P_{s,t_j}, $t_j \in E^i$.
2. Note that there could be candidate end points in E^i that are close to P_{s,t_j}. These candidate end points are considered redundant and can be removed. These redundant end points are removed by $E^{i+1} = E^i / (P_{s,t_j} \oplus T)$, where $\oplus$ is the Minkowski addition and T is a template structure. When $E^i = \phi$, the algorithm terminate.

2.2.3 Polygonal path approximation

Since the approximated centerline derived by applying the awarding functions is not smooth, (see Fig. 1(c)), we made an approximated polygonal path for each branch and at the meanwhile, the branch points are preserved. The approximation method is described below.

Given a polygonal path $S = \langle v_0, \ldots, v_m \rangle$ and an error bound ε, we look for a polygonal path, $\hat{S}$ that is a ε-approximation of S. $\hat{S} = \langle u_0, \ldots, u_m \rangle$ optimally ε-approximates S if $\hat{S}$ meets the following criteria.

1. Vertex set of $\hat{S}$ is a subset of S.
2. let $u_i = v_j$ and $u_{i+1} = v_k$, $i = 1, \ldots, m-1$, the distance between any vertex on the polygonal path $\langle v_j, \ldots, v_k \rangle$ to the line segment (u_i, u_{i+1}) is less than ε.
3. The number of the vertices on $\hat{S}$ is the least possible.

This problem can be solved by using the dynamic programming technique.

We define the number of edges on $\hat{S}_{(i,j)}$ to be its cost. The least cost among all the ε-approximation for S is the optimal cost denoted $c(i, j)$. For the boundary condition that

$i = j$, we let $c(i, j) = 1$. If $j > i$, there are two cases to establish the optimal ε-approximation path.

Case 1 : $\hat{S}_{(i\,j)}$ is the line segment (v_i, v_j)

This case occurs when all the distances between vertices v_k, $i \le k \le j$, to (v_i, v_j) are less than ε. (v_i, v_j) ε-approximates $<v_i, v_{i+1}, \ldots, v_j>$ and thus $c(i, j) = 1$.

Case2: $\hat{S}_{(i,\,j)}$ consists of two or more line segments.

In this case, $\hat{S}_{(i,\,j)}$ can be divided into two sub-path $\hat{S}_{(i\,,k)}$ and $\hat{S}_{(k,\,j)}$ where v_k is a vertex on $<v_i, \ldots, v_j>$. Note that both $\hat{S}_{(i\,,k)}$ and $\hat{S}_{(k,\,j)}$ ε-approximate polygonal paths $<v_i, \ldots, v_k>$ and $<v_k, \ldots, v_j>$. The cost for optimal ε-approximation $c(i, j)$ is $\min_{i<k<j}(c(i,k)+c(k,j))$

Based on the above discussion, the optimal cost can be written in the recurrence.

$$
\begin{aligned}
&c(i, i) = 1\\
&c(i, j) = 1 \quad \textit{if } (v_i, v_j) \textit{ ε-approximates } <v_i, \ldots, v_j> \qquad (4)\\
&c(i, j) = \min_{i<j<k}(c(i,k)+c(k,j))
\end{aligned}
$$

And the optimal solution is obtained in a bottom-up manner.

3. EXPERIMENT AND RESULTS

3.1 The data sets

All of the data were acquired in the Brain Research Center, National Tsing Hua University, Hsin-Chu with a Zeiss LSM 510 confocal microscope. The mean voxel sizes for a *Drosophila's* projection neuron image stack are $0.17\times0.17\times1.0\mu m^3$

3.2 Evaluation of the proposed method

In our experiment, we set $\lambda = 1.0$, $w = 1.0$ and the template T, is a box with sizes $5\times5\times3$ and the awarding functions is simply

$$f(p,q) = -0.5 \qquad \text{if} \quad q \in S_k \ \text{ for some } k. \qquad (5)$$

Although this awarding function looks naïve but the result is good. In Fig. 1, we present the maximum intensity projection (MIP) of the original data and the traced result when the awarding function is applied. Fig. 1(d) shows the ε-approximated centerline. In Fig. 2, the ε-approximated centerline of a more complicated structure, the projection neuron in *Drosophila's* lateral horn is presented.

.

4. DISCUSSION AND CONCLUSION

In this paper we present a practical neuron tracking algorithm based on minimal path finding method. This algorithm performs well when we deal with a 3-D image stack in which the spatial resolution in one direction is much worse than the spatial resolutions in the other two directions. To process this kind of data, we propose to separately compute the skeleton of every object in the foreground of the image stack slice-by-slice first. The candidate end points are then calculated from each slice. There are over-estimated candidate end points. The redundant end points can be removed during the construction of the 3-D shortest paths. The proposed method avoids human intervention to choose the neuron tips as end points in 3-D volume. Moreover, instead of only considering the path length, the designated awarding function keeps the traced line structure lies in the middle of the desired structure. In conclusion, the proposed algorithm is promising to trace the neurons in spite of the insufficient resolution in one direction of the acquired images.

5. REFERENCES

[1] H.H. Lin, J.S. Lai, A.L. Chin, Y.C. Chen, and A.S. Chiang, "A map of olfactory representation in the Drosophila mushroom body," *Cell*, 128(6):1205-17, 2007 Mar 23.

[2] G. S. Jefferis, C. J. Potter, A.M. Chan, E.C. Marin, T. Rohlfing, C.R. Maurer Jr, and L. Luo, "Comprehensive maps of Drosophila higher olfactory centers: spatially segregated fruit and pheromone representation," *Cell.,* 128(6):1187-203, 2007 Mar 23.

[3] S. Bouix, K. Siddiqi, and A. Tannenbaum, "Flux driven automatic centerline extraction," *Medical Image Analysis.,* Vol. 9, pp. 209-221, 2005.

[4] M. Sonka, M. D. Winniford, X. Zhang, and S. M. Collins, "Lumen centerline detection in complex coronary angiogram," *IEEE Tran. Med. Imaging,* Vol. 41, pp. 520-528, 1994.

[5] Khalid A. Al-Kofahi, S. Lasek, D. H. Szarowski, C. J. Pace, G. Nagy, and J. N. Turner, "Rapid automated three-dimensional tracing of neurons from confocal image stacks," *IEEE Tran. Info. Tech. in Biomedicine,* Vol. 6, No.2, pp. 171-187, Jun. 2002.

[6] T. Deschamps and Laurent D. Cohen, "Fast extraction of minimal paths in 3D images and application to virtual endoscopy," *Medical Image Analysis,* Vol. 5, pp. 281-299, 2001.

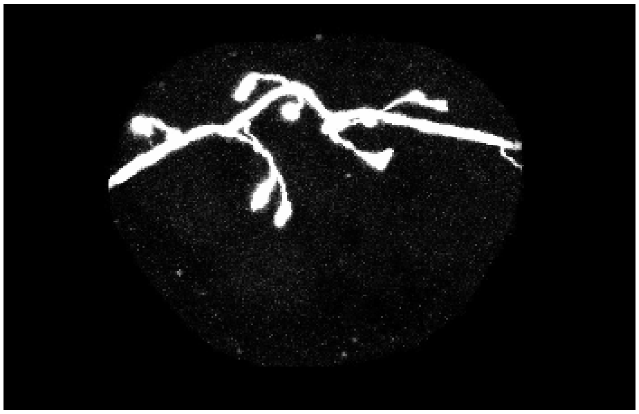
[7] I. Bitter, Arie E. Kaufman and M. Sato, "Penalized-distance volumetric skeleton algorithm," *IEEE Tran. Visualization and Computer Graphics,* Vol. 7, No. 3, pp. 195-205, Jul-Sep., 2001

[8] H. Li and A. Yezzi, "Vessels as 4-D curves: global minimal 4-D paths to extract 3-D tubular surfaces and centerlines," *IEEE Tran. Med. Imaging,* Vol. 26, No. 9, pp. 1213- 1223, Sep., 2007

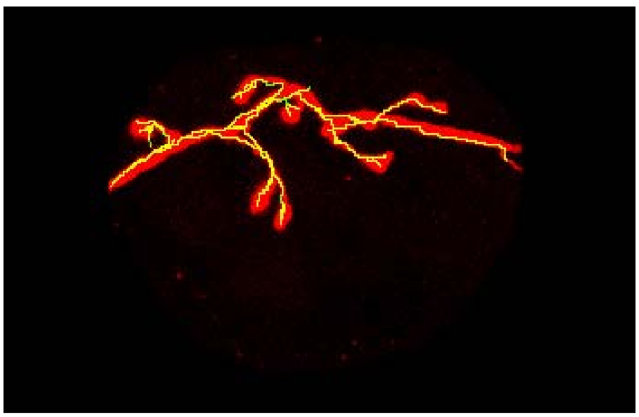
[9] Laurent D. Cohen and R. Kimmel, "Global minimum for active contour models: a minimal path approach," *Journal of Computer Vision,* 24(1), pp. 57-78, 1997

[10] M. Kass, A. Witkin and D. Terzopoulos, "Snake: active contour models," *Intl. Journal of Computer Vision,* 1(4), pp. 321-331, 1987

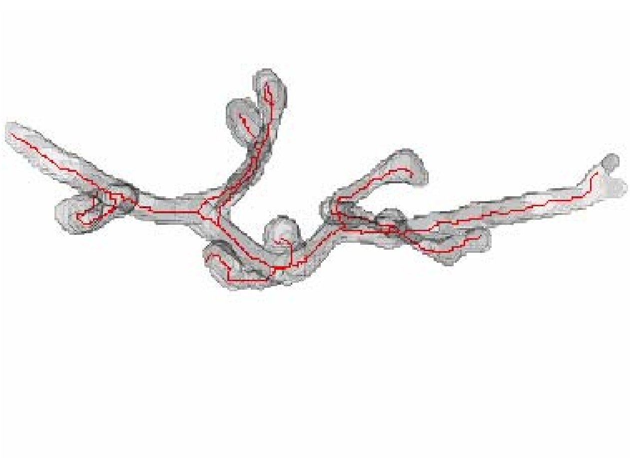
[11] C. Pudney "Distance-ordered homotopic thining: a skeletonization algorithm for 3D digital images," *Computer Vision and Image Understanding,* 72(3), pp. 404-413, 1998



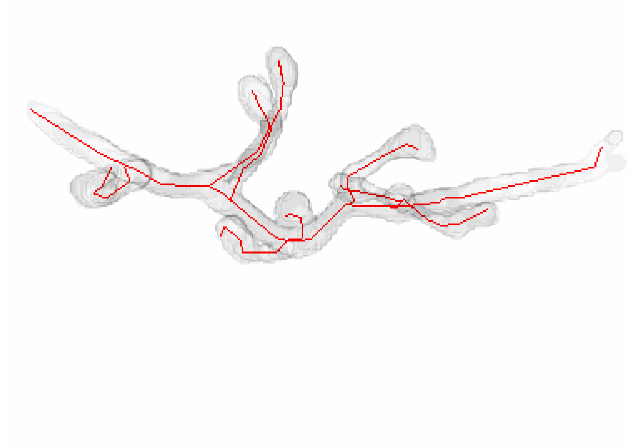
(a)



(b)



(c)



(d)

Fig. 1. The projection neuron in *Drosophila's* calyx (a) The maximum intensity projection (MIP) of the original image stack. (b) The neuron tracing result without using the rewarding function. The tracing result is drawn in green and is overlap to the MIP of the original data which is drawn in red (c) Red lines shows the tracing result with the rewarding function is applied. (d) The ε-approximation of the tracing result shows in (c) and $\varepsilon = \sqrt{2}$. Both (c) and (d) are rendered by directed volume rendering. The voxel size is $0.18 \times 0.18 \times 1.0 \mu m^3$

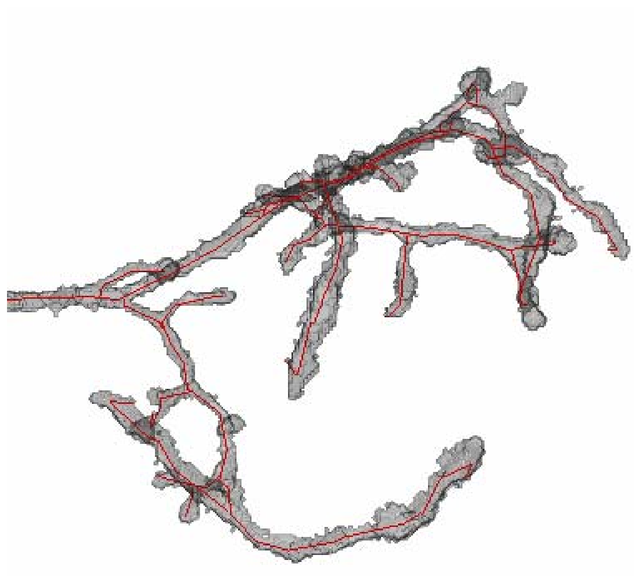


Fig. 2. This figure shows the ε-approximation, $\varepsilon = \sqrt{2}$, of the *Drosophila*'s projection neuron in the lateral horn. The red lines are tracking results. This figure is render by directed volume rendering. The voxel size is $0.16 \times 0.16 \times 1.0 \mu m^3$